Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A logic verification system comprising:

a first logic cone extraction section for extracting first logic cones from a machine-executable object code compiled from a behavioral level description written in a programming language, the object code being used for specification verification by simulation on a CPU in a design phase; and

~~wherein,~~ means for determining, based on the logic cones, whether logic circuits that have been designed in a behavioral synthesis phase are acceptable to be used in a manufacturing phase for the logic circuits.

2. (Previously Presented) The logic verification system according to claim 1, further comprising:

a second logic cone extraction section for extracting second logic cones from an RT level description; and

a logic cone comparison section for comparing the first logic cones and the second logic cones to verify equivalence between the first and second logic cones,

wherein, based on the comparison of the first logic cones and the second logic cones, the determining means determines whether the RT level description that has been designed in the behavioral synthesis phase is acceptable to be used in a manufacturing phase for the logic circuits.

3. (Original) The logic verification system according to claim 1, wherein the first logic cone extraction section includes a symbolic simulation section.

4. (Original) The logic verification system according to claim 2, wherein the first logic cone extraction section includes a symbolic simulation section.

5. (Previously Presented) A logic verification system comprising:

a storage section for storidng an object code compiled from an behavioral level description written in a programming language, the object code being used for specification verification by simulation on a CPU in a design phase,

an RT level description generated from the behavioral level description,

correspondence information which specifies information on pairs of fragments of descriptions to be compared which are included in the behavioral level description and the RT level description and which specifies information on pairs of signals to be compared for each description pair, and

compile information including mapping information between the behavioral level description and the object code;

a first logic cone extraction section for extracting first logic cones of variables by:

searching a code portion and the variables of the object code corresponding to each fragments of descriptions and each signals of behavioral level description to be compared which are specified by the correspondence information by referencing the compile information,

setting initial symbol values in the variables,

performing symbolic simulation from the start to end points of the code portion to produce symbol values when the variable symbolic simulation ends, and

using the symbol values as the first logic cones of the variables;

a second logic cone extraction section for extracting second logic cones each for the signals for each fragments of description of RT level description to be compared which are specified by the correspondence information;

a logic cone comparison section for comparing the first logic cones and the second logic cones for each signals for each of the fragments of descriptions to be compared in the behavioral level description and the RT level description which are specified by the correspondence information; and

means for determining, based on the comparison of the first logic cones and the second logic cones, whether the RT level description that has been designed in a behavioral synthesis phase is acceptable to be used in a manufacturing phase for the logic circuits.

6. (Previously Presented) A logic verification system comprising:

a first logic cone extraction section for extracting first logic cones from a machine-executable object code compiled from an behavioral level description written in a

programming language, the object code being used for specification verification by simulation on a CPU in a design phase;

a storage section for storing properties to be met by the behavioral level description;

a model checking section for checking whether the object code meets the properties; and

means for determining, based on the logic cones, whether logic circuits that have been designed in a behavioral synthesis phase are acceptable to be used in a manufacturing phase for the logic circuits.

7. (Previously Presented) A logic cone extraction apparatus comprising:

an input section for inputting an object code compiled from a program description, correspondence information which specifies logic cone extraction areas within the program description and signals to be extracted for each of the logic cone extraction areas, and compile information including mapping information between the program description and the object code, the object code being used for specification verification by simulation on a CPU in a design phase;

a symbolic simulation section which, by referencing the compile information, searches a code portion and variables of the object code corresponding to logic cone extraction areas and signals to be extracted which are specified by the correspondence information, sets initial symbol values in the variables, and performs symbolic simulation from the start to end points of the code portion;

an output section for outputting symbol values which are obtained when the variable symbolic simulation ends, as logic cones of the variables; and

means for determining, based on the logic cones, whether the logic circuits that have been designed in the design phase are acceptable to be used in a manufacturing phase for the logic circuits.

8. (Previously Presented) A logic verification method comprising the step of extracting first logic cones from a machine-executable object code compiled from a behavioral level description written in a programming language, the object code being used for specification verification by simulation on a CPU in a design phase; and

means for determining, based on the logic cones, whether logic circuits that have been designed in the design phase are acceptable to be used in a manufacturing phase for the logic circuits.

9. (Previously Presented) The logic verification method according to claim 8, further comprising the steps of:

extracting second logic cones from an RT level description; and

comparing the first logic cones and the second logic cones to verify equivalence between the first and second logic cones,

wherein, based on the comparison of the first logic cones and the second logic cones, the determining means determines whether the RT level description that has been designed in a behavioral synthesis phase is acceptable to be used in a manufacturing phase for the logic circuits.

10. (Original) The logic verification method according to claim 8, wherein the first logic cones are extracted by performing symbolic simulation.

11. (Original) The logic verification method according to claim 9, wherein the first logic cones are extracted by performing symbolic simulation.

12. (Previously Presented) A logic verification method comprising the steps of:

inputting an object code compiled from an behavioral level description written in a programming language, an RT level description generated from the behavioral level description, correspondence information which specifies information on pairs of fragments of descriptions to be compared which are included in the behavioral level description and the RT level description and which specifies information on pairs of signals to be compared for each description pair, and compile information including mapping information between the behavioral level description and the object code, the object code being used for specification verification by simulation on a CPU in a design phase;

searching a code portion and the variables of the object code corresponding to each fragments of description and each signals of behavioral level description to be compared

which are specified by the correspondence information by referencing the compile information;

setting initial symbol values in the variables;

performing symbolic simulation from the start to end points of the code portion;

determining first logic cones of the variables as symbol values when the variable symbolic simulation ends;

extracting second logic cones each for the signals for each fragments of RT level description to be compared which are specified by the correspondence information;

comparing the first logic cones and the second logic cones for each signals for each of the descriptions to be compared in the behavioral level description and the RT level description which are specified by the correspondence information; and

determining, based on the comparing step, whether the RT level description that has been designed in a behavioral synthesis phase is acceptable to be used in a manufacturing phase for the logic circuits.

13. (Previously Presented) A logic verification method comprising the steps of:

extracting first logic cones from a machine-executable object code compiled from an behavioral level description written in a programming language, the object code being used for specification verification by simulation on a CPU in a design phase;

inputting properties to be met by the behavioral level description;

checking whether the object code meets the properties; and

determining, based on the logic cones, whether logic circuits that have been designed in the design phase are acceptable to be used in a manufacturing phase for the logic circuits.

14. (Previously Presented) A logic cone extraction method comprising the steps of:

inputting an object code compiled from a program description, correspondence information which specifies logic cone extraction areas within the program description and signals to be extracted for each of the logic cone extraction areas, and compile information including mapping information between the program description and the object code, the object code being used for specification verification by simulation on a CPU in a design phase;

searching a code portion and variables of the object code corresponding to logic cone extraction areas and signals to be extracted which are specified by the correspondence information by referencing the compile information;

setting initial symbol values in the variables;

performing symbolic simulation from the start to end points of the code portion; and

outputting symbol values which are obtained when the variable symbolic simulation ends, as logic cones of the variables; and

determining, based on the logic cones, whether the logic circuits that have been designed in the design phase are acceptable to be used in a manufacturing phase for the logic circuits.

15. (Previously Presented) A computer program product embodied in computer-readable medium and comprising code that, when executed, causes a computer to perform logic verification, the program product comprising the steps of:

a) extracting first logic cones from a machine-executable object code compiled from a behavioral level description written in a programming language, the object code being used for specification verification by simulation on a CPU in a design phase;

b) extracting second logic cones from an RT level description;

c) comparing the first logic cones and the second logic cones to verify equivalence between the first and second logic cones; and

d) determining, based on the comparison of the first logic cones and the second logic cones, whether the RT level description that has been designed in a behavioral synthesis phase is acceptable to be used in a manufacturing phase for the logic circuits.

16. (Previously Presented) The computer program product according to claim 15, wherein, in the step a), the first logic cones are extracted by performing symbolic simulation.

17. (Previously Presented) A computer program product embodied in computer-readable medium and comprising code that, when executed, causes a computer to perform logic verification, the program product comprising the steps of:

a) storing an object code compiled from an behavioral level description written in a programming language, the object code being used for specification verification by

simulation on a CPU in a design phase, an RT level description generated from the behavioral level description, correspondence information which specifies information on pairs of fragments of descriptions to be compared which are included in the behavioral level description and the RT level description and which specifies information on pairs of signals to be compared for each description pair, and compile information including mapping information between the behavioral level description and the object code;

b) extracting first logic cones from a machine-executable object code compiled from a behavioral level description written in a programming language;

c) extracting second logic cones from an RT level description;

d) comparing the first logic cones and the second logic cones to verify equivalence between the first and second logic cones; and

e) determining, based on the comparison of the first logic cones and the second logic cones, whether the RT level description that has been designed in a behavioral synthesis phase is acceptable in a manufacturing phase for the logic circuits,

wherein the step b) comprises the steps of:

b.1) searching a code portion and the variables of the object code corresponding to each fragments of description and each signals of behavioral level description to be compared which are specified by the correspondence information by referencing the compile information;

b.2) setting initial symbol values in the variables;

b.3) performing symbolic simulation from the start to end points of the code portion; and

b.4) determining the first logic cones of the variables as symbol values when the variable symbolic simulation ends.

18. (Previously Presented) The computer program product according to claim 17, wherein

the step b) comprises the step of extracting the second logic cones each for the signals for each fragments of RT level description to be compared which are specified by the correspondence information, and

the step c) comprises the step of comparing the first logic cones and the second logic cones for each signals for each of the fragments of descriptions to be compared in the

behavioral level description and the RT level description which are specified by the correspondence information.

19. (Previously Presented) A computer program product embodied in computer-readable medium and comprising code that, when executed, causes a computer to perform logic verification, the program product comprising the steps of:

extracting first logic cones from a machine-executable object code compiled from an behavioral level description written in a programming language, the object code being used for specification verification by simulation on a CPU in a design phase;

inputting properties to be met by the behavioral level description;

checking whether the object code meets the properties based on the first logic cones; and

determining whether the logic circuits that have been designed in the design phase are acceptable to be used in a manufacturing phase for the logic circuits.

20. (Currently Amended) A computer program product embodied in computer-readable medium and comprising code that, when executed, causes a computer to perform logic cone extraction, the logic cone extraction comprising the steps of:

inputting an object code compiled from a program description, correspondence information which specifies logic cone extraction areas within the program description and signals to be extracted for each of the logic cone extraction areas, and compile information including mapping information between the program description and the object code, the object code being used for specification verification by simulation on a CPU in a design phase;

searching a code portion and variables of the object code corresponding to logic cone extraction areas and signals to be extracted which are specified by the correspondence information by referencing the compile information;

setting initial symbol values in the variables;

performing symbolic simulation from the start to end points of the code portion;

outputting symbol values which are obtained when the variable symbolic simulation ends, as logic cones of the variables; and

determining, based on the ~~the~~ symbol values, whether the logic circuits designed in the design phase are acceptable to be used in a manufacturing phase for the logic circuits.

21. (Previously Presented) A logic cone extraction apparatus comprising:

a first storage section for storing correspondence information which species logic cone extraction areas within a program;

a first data processing device for compiling an object code from a program description, the object code being used to describe logic circuits in a design phase for the logic circuits;

a second storage section for storing the compiled object code output by the first data processing device, and compile information;

a second data processing device for receiving program code describing logic cones, and for receiving the complied object code and the compile information stored in the second storage section, the second data processing device computing and outputting behavioral level logic cones and RT level logic cones; and

a third storage section for storing the behavioral level logic cones and RT level logic cones output by the second data processing device;

a third data processing device for receiving the behavioral level logic cones and the RT level logic cones stored in the third storage section, the correspondence information stored in the first storage section, and the compile information stored in the second storage section, and for performing logic cone comparisons as a result thereof; and

determining means for determining, based on the comparisons of the logic cones performed by the third data processing device, whether the logic circuits that have been designed in the design phase are acceptable to be used in a manufacturing phase for the logic circuits.